mathematics

MDPI

Article

Generalized Traub Family for Solving Nonlinear Systems: Fourth-Order Optimal Method and Dynamical Analysis

Alicia Cordero [1], Miguel A. Leonardo Sepúlveda [2,3,*], Juan R. Torregrosa [1], Antmel Rodríguez Cabral [4] and María P. Vassileva [5]

[1] Instituto de Matemática Multidisciplinar, Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia, Spain; acordero@mat.upv.es (A.C.); jrtorre@mat.upv.es (J.R.T.)
[2] Departamento de Matemática, Instituto Superior de Formación Docente Salomé Ureña (ISFODOSU), Av. Caonabo, Santo Domingo 10114, Dominican Republic
[3] Departamento de Matemática, Universidad APEC (UNAPEC), Avenida Máximo Gómez No. 72, Sector El Vergel, Santo Domingo 10107, Dominican Republic
[4] Escuela de Matemáticas, Universidad Autónoma de Santo Domingo (UASD), Ciudad Universitaria, Av. Alma Mater, Santo Domingo 10105, Dominican Republic; arodriguez05@uasd.edu.do
[5] Ciencias Básicas y Ambientales (CBA), Instituto Tecnológico de Santo Domingo (INTEC), Avda. Los Próceres, Santo Domingo 10602, Dominican Republic; maria.penkova@intec.edu.do
* Correspondence: mleonardos@unapec.edu.do or miguel.leonardo@isfodosu.edu.do

Abstract

A novel two-stage procedure for approximating solutions of nonlinear systems is introduced. The scheme employs two evaluations of the vector function F together with a single Jacobian computation, followed by the resolution of two linear subproblems that share an identical coefficient matrix. This structure reduces the computational cost and enhances the adaptability of the method with respect to existing alternatives. The design of the algorithm is motivated by criteria relating efficiency to the total number of functional evaluations, ensuring that the resulting strategy achieves the optimal convergence order permitted within this framework. A proof of the local convergence order is provided, and its accuracy is supported by a series of experiments on distinct nonlinear models, including problems arising from differential equations. The numerical evidence confirms that the developed technique reaches the theoretical convergence rate and performs favorably when compared with other methods of equal order. Moreover, we examine the dynamical features of the related parametric variant, offering additional understanding of its stability properties and iterative behavior.

Keywords: nonlinear systems; iterative methods; Generalized Traub; high-order iterative schemes; dynamical analysis

MSC: 65H10; 37N30

Check for updates

Academic Editor: Jonathan Blackledge

Received: 2 March 2026
Revised: 23 March 2026
Accepted: 27 March 2026
Published: 31 March 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

The numerical approximation of solutions to nonlinear systems continues to play a central role in computational mathematics. Let $F : \mathcal{D} \subseteq \mathbb{R}^n \to \mathbb{R}^n$ be a sufficiently smooth mapping, and consider the nonlinear system

$$F(x) = 0.$$

Iterative procedures are usually employed to approximate its solutions when an explicit closed-form expression is either unavailable or unsuitable for practical computation. New-

ton's classical scheme, which combines quadratic convergence with a minimal distribution of evaluations between the vector function F and its Jacobian F', is widely regarded as the first optimal second-order procedure under the scalar Kung–Traub conjecture. This optimality interpretation was later extended to systems of equations. Nevertheless, the intrinsic restriction to order two has stimulated the search for higher-order frameworks that preserve optimality while complying with known efficiency limitations.

Inspired by the scalar Kung–Traub setting, Cordero–Torregrosa [1] established an optimality conjecture for nonlinear systems (without memory). According to this conjecture, a scheme using k_1 Jacobian evaluations and k_2 evaluations of the vector function per iteration, with $k_1 \leq k_2$, cannot surpass the upper limit $2^{k_1+k_2-1}$. Consequently, the conjecture plays for systems the same structural role that Kung–Traub plays in the scalar case. Under this criterion, Newton's method is confirmed as an optimal second-order algorithm, whereas achieving *fourth order* with a feasible cost distribution (k_1, k_2) constitutes the next theoretical target.

In the way towards higher-order optimality, several efficient (but still not optimal) schemes have been designed through the use of weight functions and corrective mechanisms. In particular, Sharma et al. [2] introduced the squared-norm ratio

$$\mu_k = \frac{\|F(y^{(k)})\|_2^2}{\|F(x^{(k)})\|_2^2}$$

which acts as a refinement term capable of improving the convergence order without requiring additional functional evaluations. This idea has since been incorporated into various multidimensional environments.

A significant contribution toward this goal was recently given by Cordero et al. [3], who developed the first optimal fourth-order family for nonlinear systems by generalizing the Ermakov hyperfamily to the multidimensional setting. Their framework provides an efficient vectorial extension of several well-known scalar schemes, including Ostrowski's method [4], King's family [5], and Chun's method [6], as well as KLAM-type variants, while incorporating the quantity μ_k as a weighting factor. Their results include a rigorous proof of fourth-order optimal convergence under the Cordero–Torregrosa conjecture, together with extensive numerical verification.

Despite these advances, the class of optimal methods for nonlinear systems remains very limited. In particular, only a few recent contributions have achieved optimality in the sense of the Cordero–Torregrosa conjecture. For instance, Cordero et al. [3] obtained the first optimal fourth-order family for nonlinear systems, while Cordero et al. [7] proposed the first optimal eighth-order method in this framework. This scarcity reveals a clear research gap and highlights the need for new optimal families that, in addition to satisfying the conjectured efficiency bound, preserve a computational structure that is practical, stable, and competitive in multidimensional settings.

It must be noted, however, that extending optimal scalar procedures to the multidimensional case does not always preserve optimality. For example, Budzko et al. [8], following the ideas of Ermakov–Kalitkin, proposed a one-parameter PMS family of cubic order based on divided-difference operators, illustrating that the design of stable multidimensional corrections requires a careful balance in the distribution of evaluations of F and F'.

Conversely, although Jarratt's method has optimal fourth-order in one dimension, its extension to systems, analyzed by Khirallah et al. [9], fails to be optimal due to its high computational cost ($k_1 > k_2$).

In light of these considerations, we introduce a new class of iterative schemes for solving nonlinear systems, which we name Generalized Traub. The proposed construction (i) employs the quotient μ_k to accelerate the convergence without adding new functional

evaluations; (ii) incorporates a parametrization that ensures optimality according to the Cordero–Torregrosa framework; and (iii) maintains a competitive cost per iteration by requiring exactly two evaluations of F, one of a single Jacobian matrix, and the solution of two linear systems sharing the same coefficient matrix. Our design takes the classical third-order Traub method as its core and generates the parametric family Generalized Traub with parameter $\gamma \in \mathbb{R}$. In general, the family holds third order of convergence; however, for the special value $\gamma = 2$ we obtain a fourth-order member. Since this variant uses $(k_1, k_2) = (1, 2)$, the Cordero–Torregrosa upper bound becomes $2^{k_1+k_2-1} = 2^2 = 4$, implying that this method is therefore optimal under the conjecture.

Beyond its theoretical interest, the proposed family may also be relevant in broader computational settings arising in related areas where nonlinear models and repeated iterative updates play a central role. For instance, recent developments in iterative learning control for feedback systems with varying tasks and in fast finite-time consensus protocols for high-order nonlinear multi-agent systems illustrate the increasing importance of robust and efficient computational strategies in nonlinear control environments; see, for example [10,11]. In this broader context, the Generalized Traub family may be of potential interest whenever the underlying numerical treatment requires the repeated solution of nonlinear systems with a favorable balance between accuracy, stability, and computational cost.

The theoretical study provided in this work includes a detailed local convergence analysis. We also undertake a dynamical analysis of the parametric family, offering complementary insight into stability and qualitative iterative behavior. Finally, the numerical assessment conducted on benchmark and large-scale systems demonstrates that Generalized Traub is highly competitive and, in many cases, outperforms the vectorial extension of classical methods such as Ostrowski, Chun, or Jarratt with respect to stability, convergence domains, and computational performance.

Now, we summarize the definitions and known results needed in the rest of the manuscript.

1.1. Preliminary Concepts

Throughout this work, we assume that the system possesses a solution $\bar{x} \in \mathcal{D}$, such that

$$F(\bar{x}) = 0, \qquad F'(\bar{x}) \text{ is nonsingular,}$$

and that all derivatives required for the subsequent analysis exist and remain continuous in a neighborhood of $\bar{x}$.

For $m \geq 1$, the m-th derivative of the map F at a point x is the m-linear operator

$$F^{(m)}(x) : \underbrace{\mathbb{R}^n \times \cdots \times \mathbb{R}^n}_{m \text{ slots}} \longrightarrow \mathbb{R}^n, \qquad F^{(m)}(x) \in \mathcal{L}_m(\mathbb{R}^n; \mathbb{R}^n).$$

For brevity, we employ the compact notations

$$F^{(m)}(x)(\omega_1, \ldots, \omega_m) = F^{(m)}(x)\,\omega_1 \cdots \omega_m, \qquad F^{(m)}(x)\,\omega^{m-1} F^{(p)}(x)\,\omega^p = F^{(m)}(x) F^{(p)}(x)\,\omega^{m+p-1}.$$

Let $x = \bar{x} + \eta$, with η sufficiently small. Assuming that $F'(\bar{x})$ is invertible, a Taylor expansion around $\bar{x}$ yields

$$F(\bar{x} + \eta) = F'(\bar{x})\Big[\eta + D_2\eta^2 + D_3\eta^3 + D_4\eta^4\Big] + \mathcal{O}\Big(\eta^5\Big), \tag{1}$$

where the tensors D_j are given by

$$D_j = \frac{1}{j!}\,[F'(\bar{x})]^{-1} F^{(j)}(\bar{x}), \qquad j \geq 2.$$

A similar expansion for the Jacobian operator of F at $\bar{x}+\eta$ around $\bar{x}$ reads

$$F'(\bar{x}+\eta) = F'(\bar{x})\Big[I + 2D_2\eta + 3D_3\eta^2 + 4D_4\eta^3\Big] + \mathcal{O}\Big(\eta^4\Big). \tag{2}$$

Moreover, the inverse matrix $[F'(\bar{x}+\eta)]^{-1}$ admits the Neumann-type expansion

$$\begin{aligned}\big[F'(\bar{x}+\eta)\big]^{-1} = \Big[&I - 2D_2\eta + (4D_2^2 - 3D_3)\eta^2 \\ &+ \big(-8D_2^3 + 6D_3D_2 + 6D_2D_3 - 4D_4\big)\eta^3\Big][F'(\bar{x})]^{-1} + \mathcal{O}\Big(\eta^4\Big).\end{aligned} \tag{3}$$

Given a sequence of approximations $\{x^{(k)}\}$ to $\bar{x}$, we define the local error at iteration k by

$$\varepsilon^{(k)} = x^{(k)} - \bar{x}.$$

Then, the convergence of order p of sequence $\{x^{(k)}\}$ can be expressed through the *error equation*

$$\varepsilon^{(k+1)} = \mathcal{T}\,\varepsilon^{(k)\,p} + \mathcal{O}\Big(\varepsilon^{(k)^{p+1}}\Big),$$

where $\mathcal{T}$ denotes a suitable p-linear operator characterizing the leading term of the local behavior.

For numerical estimation, the order of convergence p is computationally approximated by

$$p \approx ACOC = \frac{\ln\Big(\|x^{(k+1)} - x^{(k)}\| \,/\, \|x^{(k)} - x^{(k-1)}\|\Big)}{\ln\big(\|x^{(k)} - x^{(k-1)}\| \,/\, \|x^{(k-1)} - x^{(k-2)}\|\big)}. \tag{4}$$

1.1.1. Cost Model and Efficiency Indices

To classify the iterative schemes in terms of efficiency, the number of functional evaluations d and products/quotients op per iteration is taken into account. The classical Ostrowski's efficiency index is $E = p^{1/d}$ [4]; meanwhile, Traub's operational index is $C = p^{1/op}$ [12]. To properly calculate them, it must be taken into account that if, in one iteration, we solve m linear systems with the same coefficient matrix by LU factorization (once), a standard dense model gives

$$\tfrac{1}{3}n^3 + mn^2 - \tfrac{1}{3}n$$

products/quotients. Following [13], we also report the *computational efficiency index* $CE = p^{1/(d+op)}$, combining functional and operational costs.

Conjecture 1 (Cordero–Torregrosa [1]). *Let $F(x) = 0$ be a nonlinear system, and consider an iterative procedure without memory designed to approximate its solutions. Suppose that, at each iteration, the scheme requires k_1 evaluations of the Jacobian matrix $F'(x)$ and k_2 evaluations of the map $F(x)$, where $k_1 \leq k_2$. Then, the attainable local order of convergence p is bounded by*

$$p \;\leq\; 2^{k_1+k_2-1}.$$

The method is said to be optimal precisely when this bound is attained, i.e., when $p = 2^{k_1+k_2-1}$.

1.1.2. Concepts from Multidimensional Real Dynamics

Discrete dynamical systems provide a powerful viewpoint for examining the qualitative behavior of iterative schemes applied to nonlinear problems. A common strategy for studying stability consists of analyzing the rational operator induced by the method when it is applied to polynomial systems of low degree (typically quadratic or cubic), as discussed in [14,15].

In the univariate case, such analysis is often carried out in the realm of real or complex dynamics. In contrast, the present work deals with vector-valued iterative families, which require a multidimensional real dynamical perspective. To that end, we consider quadratic polynomial systems and apply our scheme to obtain the corresponding multidimensional rational operator. This resulting operator is then examined through its fixed, periodic, and critical points, which allows us to identify parameter values leading to favorable stability outcomes.

We briefly recall some notions employed throughout this study, which are extensions of the concepts for the scalar complex case (see, for instance, [16]). Let $G : \mathbb{R}^n \to \mathbb{R}^n$ denote the rational operator obtained after applying an iterative scheme to a polynomial system $p(x)$. The sequence

$$\{x^{(0)}, G(x^{(0)}), G^2(x^{(0)}), \ldots, G^m(x^{(0)}), \ldots\},$$

is known as the orbit of the initial point $x^{(0)}$. A point $\bar{x} \in \mathbb{R}^n$ is a fixed point of G if $G(\bar{x}) = \bar{x}$. Always, the zeros of $p(x)$ are fixed points of G, but other fixed points may appear that do not correspond to actual solutions of the system; such points are called extraneous fixed points. More generally, a point x is periodic of period m if

$$G^m(x) = x, \qquad \text{with } G^{m-j}(x) \neq x \text{ for } 0 < j < m.$$

To determine the stability of fixed or periodic points, we recall the following criterion.

Theorem 1 ([17], p. 558)**.** *Let $G : \mathbb{R}^n \to \mathbb{R}^n$ be of class C^2, and suppose that $\bar{x}$ is a periodic point of period $m \geq 1$. Let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of the Jacobian matrix $G'(\bar{x})$. Then:*

1. *If $|\lambda_j| < 1$ for all j, then $\bar{x}$ is an attracting point.*
2. *If some j_0 satisfies $|\lambda_{j_0}| > 1$, then $\bar{x}$ is unstable (either repelling or of saddle type).*
3. *If $|\lambda_j| > 1$ for every j, then $\bar{x}$ is a repelling point.*

The collection of all initial values converging to an attracting fixed point $\bar{x}$ under the iteration of G is termed its *basin of attraction*:

$$\mathcal{A}(\bar{x}) = \left\{ x^{(0)} \in \mathbb{R}^n : \ G^m(x^{(0)}) \to \bar{x} \text{ as } m \to \infty \right\}.$$

Points satisfying $G'(x) = 0$ are called *critical points* of G. Critical points that are not zeros of $p(x)$ are known as *free critical points*. These objects play a fundamental role in the stability analysis, due to the classical theorem of Julia and Fatou [18–20].

Theorem 2 (Julia–Fatou)**.** *Let G be a rational function. Then, the immediate basin of attraction of an attracting periodic (or fixed) point contains at least one critical point.*

The structure of the paper is as follows. Section 2 presents the new class of iterative schemes and its convergence properties. Section 3 is devoted to the dynamical analysis of the parametric family. Section 4 contains the comparative numerical experiments. Section 5 presents the conclusions and outlines possible directions for future research.

2. Convergence Order of the Generalized Traub Methods for Systems

To introduce the methods, we define

$$\begin{cases} y^{(k)} = x^{(k)} - \left[F'(x^{(k)})\right]^{-1} F\left(x^{(k)}\right), \\ z^{(k)} = y^{(k)} - \left[F'(x^{(k)})\right]^{-1} F\left(y^{(k)}\right), \qquad k = 0, 1, 2, \ldots \\ x^{(k+1)} = z^{(k)} - \gamma\, \mu_k \left(x^{(k)} - z^{(k)}\right), \end{cases} \tag{5}$$

where $\gamma \in \mathbb{R}$ is a free parameter. The scalar quantity

$$\mu_k = \frac{\|F(y^{(k)})\|_2^2}{\|F(x^{(k)})\|_2^2}$$

acts as a correction factor in the second stage. Each iteration of (5) requires one evaluation of the Jacobian matrix F' and two evaluations of the vector function F, together with the solution of two linear systems whose coefficient matrix $F'(x^{(k)})$ is identical.

Theorem 3. *Let $F : \mathcal{D} \subset \mathbb{R}^n \to \mathbb{R}^n$ be a sufficiently smooth mapping on an open convex domain $\mathcal{D}$, and let $\bar{x} \in \mathcal{D}$ be a solution of F. Let us also assume that $F'(x)$ is nonsingular in a neighborhood of $\bar{x}$. Consider the two-stage iteration described by*

$$\begin{cases} y^{(k)} = x^{(k)} - \left[F'(x^{(k)})\right]^{-1} F\left(x^{(k)}\right), \\ z^{(k)} = y^{(k)} - \left[F'(x^{(k)})\right]^{-1} F\left(y^{(k)}\right), \qquad k = 0, 1, 2, \ldots \\ x^{(k+1)} = z^{(k)} - \gamma\, \mu_k \left(x^{(k)} - z^{(k)}\right), \end{cases}$$

where $\mu_k = \dfrac{\|F(y^{(k)})\|_2^2}{\|F(x^{(k)})\|_2^2}$.

Let us define the local error vector $\varepsilon^{(k)} = x^{(k)} - \bar{x}$. The error equation of the proposed class of iterative schemes is

$$\varepsilon^{(k+1)} = (2-\gamma) D_2^2 \varepsilon^{(k)^3} + \left[2(2-\gamma) D_2 D_3 + (3-2\gamma) D_3 D_2 - (9-4\gamma) D_2^3 + \gamma P^{-1} Q D_2^2\right] \varepsilon^{(k)^4} + \mathcal{O}\left(\varepsilon^{(k)^5}\right),$$

and the method achieves at least third order of convergence. However, if $\gamma = 2$, the theoretical order becomes four, and

$$\varepsilon^{(k+1)} = \left[-D_3 D_2 - D_2^3 + 2P^{-1} Q D_2^2\right] \varepsilon^{(k)^4} + \mathcal{O}\left(\varepsilon^{(k)^5}\right).$$

Therefore, each iteration requires $k_1 = 1$ Jacobian evaluation and $k_2 = 2$ evaluations of the function F; thus, it reaches the Cordero–Torregrosa bound $2^{k_1+k_2-1} = 4$, becoming an optimal method, which is denoted by TGM.

Proof. To analyze the local behavior of the method, we rely on the Taylor expansions of F and of the inverse Jacobian $[F'(x)]^{-1}$ about the solution $\bar{x}$. All products involving the tensors D_j and powers of $\varepsilon^{(k)}$ are understood in the multilinear sense introduced in Section 1. Formulas (1)–(3) provide the local representations of $F(x)$ and its Jacobian inverse around $\bar{x}$, where we set

$$\eta = x^{(k)} - \bar{x} = \varepsilon^{(k)}.$$

Evaluating these expansions at $x^{(k)}$, we obtain

$$F(x^{(k)}) = F'(\bar{x}) \left(\varepsilon^{(k)} + D_2\, \varepsilon^{(k)^2} + D_3\, \varepsilon^{(k)^3} + D_4\, \varepsilon^{(k)^4}\right) + \mathcal{O}\left(\varepsilon^{(k)^5}\right). \tag{6}$$

Similarly, the inverse Jacobian evaluated at $x^{(k)}$ reads

$$[F'(x^{(k)})]^{-1} = \Big(I - 2D_2\varepsilon^{(k)} + (4D_2^2 - 3D_3)\varepsilon^{(k)^2} + (-8D_2^3 + 6D_3D_2 + 6D_2D_3 - 4D_4)\varepsilon^{(k)^3} + (16D_2^4 - 12D_2^2D_3 + 9D_3^2 + 10D_2D_4 - 5D_5)\varepsilon^{(k)^4}\Big)[F'(\bar{x})]^{-1} + \mathcal{O}\Big(\varepsilon^{(k)^5}\Big). \tag{7}$$

Then,

$$\varepsilon_y^{(k)} = y^{(k)} - x = \varepsilon^{(k)} - [F'(x^{(k)})]^{-1}F(x^{(k)}). \tag{8}$$

Substituting (6) and (7) in (8), we have

$$\varepsilon_y^{(k)} = D_2\,\varepsilon^{(k)^2} + (-2D_2^2 + 2D_3)\varepsilon^{(k)^3} + (3D_4 - 4D_2D_3 - 3D_3D_2 + 4D_2^3)\varepsilon^{(k)^4} + \mathcal{O}\Big(\varepsilon^{(k)^5}\Big). \tag{9}$$

Since $\varepsilon_y^{(k)} = \mathcal{O}\Big(\varepsilon^{(k)^2}\Big)$, applying the expansion (6) at $y^{(k)}$ yields

$$\begin{aligned} F(y^{(k)}) &= F'(\bar{x})\Big(\varepsilon_y^{(k)} + D_2\,\varepsilon_y^{(k)^2} + D_3\,\varepsilon_y^{(k)^3}\Big) + \mathcal{O}\Big(\varepsilon_y^{(k)^4}\Big) \\ &= F'(\bar{x})\Big(D_2\,\varepsilon^{(k)^2} + (-2D_2^2 + 2D_3)\varepsilon^{(k)^3} + (3D_4 - 4D_2D_3 - 3D_3D_2 + 5D_2^3)\varepsilon^{(k)^4} + \mathcal{O}\Big(\varepsilon^{(k)^5}\Big)\Big). \end{aligned} \tag{10}$$

Adding (6) and (10), we find

$$\begin{aligned} F(x^{(k)}) + F(y^{(k)}) = F'(\bar{x})\Big(&\varepsilon^{(k)} + 2D_2\,\varepsilon^{(k)^2} + (-2D_2^2 + 3D_3)\varepsilon^{(k)^3}\Big) \\ &+ (4D_4 - 4D_2D_3 - 3D_3D_2 + 5D_2^3)\varepsilon^{(k)^4} + \mathcal{O}\Big(\varepsilon^{(k)^5}\Big). \end{aligned} \tag{11}$$

Multiplying this result by the inverse Jacobian $[F'(x^{(k)})]^{-1}$ and using (7), we obtain

$$\begin{aligned} x^{(k)} - z^{(k)} =& [F'(x^{(k)})]^{-1}\big(F(x^{(k)}) + F(y^{(k)})\big) \\ =& \varepsilon^{(k)} - 2D_2^2\,\varepsilon^{(k)^3} + \big(-4D_2D_3 + 9D_2^3 - 3D_3D_2\big)\varepsilon^{(k)^4} + \mathcal{O}\Big(\varepsilon^{(k)^5}\Big), \end{aligned} \tag{12}$$

and, therefore, the error at step $z^{(k)}$ is

$$\varepsilon_z^{(k)} = 2D_2^2\,\varepsilon^{(k)^3} + \big(4D_2D_3 - 9D_2^3 + 3D_3D_2\big)\varepsilon^{(k)^4} + \mathcal{O}\Big(\varepsilon^{(k)^5}\Big),$$

Now, we compute the squared–norm quotient (cf. [2])

$$\mu_k = \frac{\|F(y^{(k)})\|_2^2}{\|F(x^{(k)})\|_2^2} = \frac{\sum_{i=1}^{m} f_i^2(y^{(k)})}{\sum_{i=1}^{m} f_i^2(x^{(k)})},$$

where $F(t) = (f_1(t), \ldots, f_m(t))^T$. Around $\bar{x}$, with $R_i = f_i'(\bar{x})$ and $G_i = \frac{1}{2}f_i''(\bar{x})$, we have

$$f_i(x^{(k)}) = R_i\,\varepsilon^{(k)} + G_i\,\varepsilon^{(k)^2} + O(\varepsilon^{(k)^3}), \qquad f_i(y^{(k)}) = R_i\,\varepsilon_y^{(k)} + G_i\,\varepsilon_y^{(k)^2} + \mathcal{O}\Big(\varepsilon_y^{(k)^3}\Big).$$

Since f_i is a scalar function,

$$f_i^2(x^{(k)}) = R_i^TR_i\,\varepsilon^{(k)^2} + \big(R_i^TG_i + G_i^TR_i\big)\varepsilon^{(k)^3} + \mathcal{O}\Big(\varepsilon^{(k)^4}\Big).$$

Setting $P_i := R_i^TR_i$, $Q_i := R_i^TG_i + G_i^TR_i$, $P := \sum_i P_i = (F')^T(F')$, and $Q := \sum_i Q_i$, we obtain

$$f_i^2(x^{(k)}) = P_i\varepsilon^{(k)^2} + Q_i\varepsilon^{(k)^3} + \mathcal{O}\Big(\varepsilon^{(k)^4}\Big), \qquad f_i^2(y^{(k)}) = P_i\varepsilon_y^{(k)^2} + Q_i\varepsilon_y^{(k)^3} + \mathcal{O}\Big(\varepsilon_y^{(k)^4}\Big),$$

and therefore

$$\frac{F(y^{(k)})^T F(y^{(k)})}{F(x^{(k)})^T F(x^{(k)})} = \frac{P\,\varepsilon_y^{(k)^2} + Q\,\varepsilon_y^{(k)^3} + \mathcal{O}\left(\varepsilon_y^{(k)^4}\right)}{P\,\varepsilon^{(k)^2} + Q\,\varepsilon^{(k)^3} + \mathcal{O}\left(\varepsilon^{(k)^4}\right)}.$$

Let $\varepsilon_y^{(k)} = D_2\,\varepsilon^{(k)^2} + D_3\,\varepsilon^{(k)^3} + O(\varepsilon^{(k)^4})$ (see (9)) . A routine quotient expansion gives

$$\frac{F(y^{(k)})^T F(y^{(k)})}{F(x^{(k)})^T F(x^{(k)})} = D_2^2\varepsilon^{(k)^2} + \left(2D_2D_3 + 2D_3D_2 - 4D_2^3 - P^{-1}Q\,D_2^2\right)\varepsilon^{(k)^3} + \mathcal{O}\left(\varepsilon^{(k)^4}\right). \quad (13)$$

Using these expansions, we calculate the error equation,

$$\begin{aligned}
\varepsilon^{(k+1)} &= z^{(k)} - \gamma\mu_k\left(x^{(k)} - z^{(k)}\right)\\
&= 2D_2^2\,\varepsilon^{(k)^3} + \left(4D_2D_3 - 9D_2^3 + 3D_3D_2\right)\varepsilon^{(k)^4}\\
&\quad - \gamma\left(D_2^2\varepsilon^{(k)^2} + \left(2D_2D_3 + 2D_3D_2 - 4D_2^3 - P^{-1}QD_2^2\right)\varepsilon^{(k)^3}\right)\\
&\quad \times \left(\varepsilon^{(k)} - 2D_2^2\varepsilon^{(k)^3} + \left(-4D_2D_3 + 9D_2^3 - 3D_3D_2\right)\varepsilon^{(k)^4}\right) + \mathcal{O}(\varepsilon^{(k)^5})\\
&= (2-\gamma)D_2^2\,\varepsilon^{(k)^3}\\
&\quad + \left[2(2-\gamma)D_2D_3 + (3-2\gamma)D_3D_2 - (9-4\gamma)D_2^3 + \gamma P^{-1}QD_2^2\right]\varepsilon^{(k)^4}\\
&\quad + \mathcal{O}(\varepsilon^{(k)^5}).
\end{aligned}$$

In particular, when $\gamma = 2$, the cubic term also vanishes and the leading term is

$$\varepsilon^{(k+1)} = \left[-D_3D_2 - D_2^3 + 2P^{-1}QD_2^2\right]\varepsilon^{(k)^4} + \mathcal{O}\left(\varepsilon^{(k)^5}\right),$$

so the method reaches order 4. □

3. Iterative Methods Used for Comparison

To examine the performance of the proposed Generalized Traub formulation, we contrast it with eight iterative procedures that are widely cited in the nonlinear systems literature. These schemes are employed as reference methods in the computational study, and their corresponding iterations are stated below.

Generalized Ostrowski-type scheme (GOM) with $\lambda = 1$ ($p = 4$). This variant incorporates a parameter λ, here fixed to $\lambda = 1$, following [21].

$$\begin{cases} y^{(k)} = x^{(k)} - [F'(x^{(k)})]^{-1}F(x^{(k)}),\\ x^{(k+1)} = z^{(k)} - \left(\lambda[F'(x^{(k)})]^{-1} - 2\lambda[x^{(k)}, y^{(k)}; F]^{-1} + (1+\lambda)\left(2[x^{(k)}, y^{(k)}; F] - F'(x^{(k)})\right)^{-1}\right)F(y^{(k)}). \end{cases} \quad (14)$$

Chun's method ($p = 4$), (CM). A Newton-type construction employing divided differences [22].

$$\begin{cases} y^{(k)} = x^{(k)} - [F'(x^{(k)})]^{-1}F(x^{(k)}),\\ x^{(k+1)} = y^{(k)} - \left(3I - 2F'(x^{(k)})^{-1}[x^{(k)}, y^{(k)}; F]\right)F'(x^{(k)})^{-1}F(y^{(k)}). \end{cases} \quad (15)$$

Ostrowski's classical iteration ($p = 4$), (OCM). This formulation augments Newton's step with an additional correction [23].

$$\begin{cases} y^{(k)} = x^{(k)} - [F'(x^{(k)})]^{-1}F(x^{(k)}),\\ x^{(k+1)} = y^{(k)} - \left(2[x^{(k)}, y^{(k)}; F] - F'(x^{(k)})\right)^{-1}F(y^{(k)}). \end{cases} \quad (16)$$

Newton–Jarratt scheme ($p = 5$), (JM). A multidimensional extension of Jarratt's approach, as described in [13].

$$\begin{cases} z^{(k)} = x^{(k)} - \frac{2}{3}\left[F'\left(x^{(k)}\right)\right]^{-1} F\left(x^{(k)}\right), \\ y^{(k)} = x^{(k)} - \frac{1}{2}\left(3F'\left(z^{(k)}\right) - F'\left(x^{(k)}\right)\right)^{-1}\left(3F'\left(z^{(k)}\right) + F'\left(x^{(k)}\right)\right)\left[F'\left(x^{(k)}\right)\right]^{-1} F\left(x^{(k)}\right), \\ x^{(k+1)} = y^{(k)} - \frac{1}{2}\left(F'\left(x^{(k)}\right) + F'\left(z^{(k)}\right)\right)^{-1} F\left(y^{(k)}\right). \end{cases} \tag{17}$$

Sharma's fourth-order procedure ($p = 4$), (SM). A Newton-type modification requiring two Jacobians per step [24].

$$\begin{cases} y^{(k)} = x^{(k)} - \frac{2}{3}\left[F'\left(x^{(k)}\right)\right]^{-1} F\left(x^{(k)}\right), \\ x^{(k+1)} = x^{(k)} - \frac{1}{2}\mathcal{L}_k\left[F'\left(x^{(k)}\right)\right]^{-1} F\left(x^{(k)}\right), \end{cases} \tag{18}$$

$$\text{with } \mathcal{L}_k = -I + \frac{9}{4}\left[F'\left(y^{(k)}\right)\right]^{-1} F'\left(x^{(k)}\right) + \frac{3}{4}\left[F'\left(x^{(k)}\right)\right]^{-1} F'\left(y^{(k)}\right). \tag{19}$$

Singh's fifth-order scheme ($p = 5$), (SiM). A higher-order iterative extension of Ostrowski-type updates [2].

$$\begin{cases} y^{(k)} = x^{(k)} - [F'(x^{(k)})]^{-1} F(x^{(k)}), \\ x^{(k+1)} = y^{(k)} - \rho_k [F'(y^{(k)})]^{-1} F(y^{(k)}), \end{cases} \tag{20}$$

where

$$\rho_k = 1 + \frac{\|F(y^{(k)})\|_2^2}{\|F(x^{(k)})\|_2^2}.$$

ACTV method ($p = 5$). A fifth-order strategy introduced by Arroyo et al. in [25].

$$\begin{cases} y^{(k)} = x^{(k)} - \left[F'\left(x^{(k)}\right)\right]^{-1} F\left(x^{(k)}\right), \\ x^{(k+1)} = y^{(k)} - \left(-F'\left(x^{(k)}\right) + 5F'\left(y^{(k)}\right)\right)^{-1}\left(3F'\left(x^{(k)}\right) + F'\left(y^{(k)}\right)\right)\left[F'\left(x^{(k)}\right)\right]^{-1} F\left(y^{(k)}\right). \end{cases} \tag{21}$$

Newton's iteration ($p = 2$), (NM). Used as a baseline:

$$x^{(k+1)} = x^{(k)} - [F'(x^{(k)})]^{-1} F(x^{(k)}). \tag{22}$$

Cost Model and Presentation of Efficiency Tables

This subsection analyzes the computational cost associated with a single iteration of each iterative scheme under consideration. The total cost is decomposed into two components: the functional evaluations and the algebraic operations required to update the iterate.

The classical efficiency index is defined by

$$I = p^{1/d},$$

where p denotes the order of convergence and d represents the number of functional evaluations per iteration.

However, for nonlinear systems, the dominant computational effort typically arises from the solution of linear systems. Therefore, we adopt the computational efficiency index

$$\mathrm{CI} = p^{1/(d+\mathrm{op})},$$

where op denotes the number of scalar products and quotients required per iteration.

Throughout the analysis, the following cost model is employed:

- Each evaluation of function F requires n scalar functional evaluations, while the evaluation of F' entails n^2 scalar functional evaluations.
- A first-order divided difference $[x^{(k)}, z^{(k)}; F]$ requires $n^2 - n$ functional evaluations and n^2 quotiens.
- The direct solution of a linear system of size $n \times n$, using LU factorization, requires

$$\frac{n^3}{3} + n^2 - \frac{n}{3},$$

products/quotients.

- If m linear systems are solved with the same coefficient matrix, the associated cost becomes

$$\frac{n^3}{3} + mn^2 - \frac{n}{3}.$$

The decomposition made in Tables 1 and 2 allows for a homogeneous comparison of all schemes for different nonlinear systems of size $n \times n$, analyzes when an increase in the order of convergence p compensates for the greater work associated with algebraic operations, and identifies which component of the algorithm dominates the computational effort as n grows. In the case of dense matrices, the cubic term $\frac{1}{3}n^3$ is generally the dominant contribution.

Table 1. Convergence order and per-iteration evaluation counts (#F, #F', #DD) for the nine methods.

Method	Order	#F	#F'	#DD
GTM	4	2	1	0
GOM	4	2	1	1
CM	4	2	1	1
OCM	4	2	1	1
JM	5	2	2	0
SM	4	1	2	0
SiM	5	2	2	0
ACTV	5	2	2	0
NM	2	1	1	0

Table 2. Per-iteration costs (op, d) and efficiency index $p^{1/(d+\mathrm{op})}$ for the nine methods.

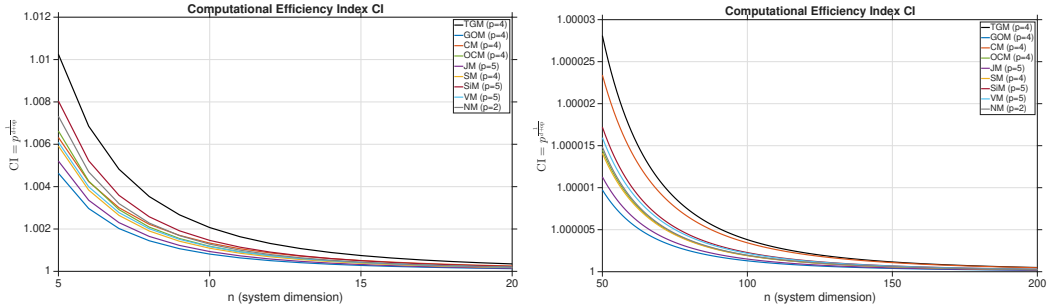
Method	op	d	$p^{1/(d+\mathrm{op})}$
GTM	$\frac{1}{3}n^3 + 2n^2 - \frac{5}{3}n + 1$	$n^2 + 2n$	$4^{\frac{1}{\frac{1}{3}n^3+3n^2+\frac{11}{3}n+1}}$
GOM	$n^3 + 5n^2 - n$	$2n^2 + n$	$4^{\frac{1}{n^3+7n^2}}$
CM	$\frac{1}{3}n^3 + 5n^2 - \frac{1}{3}n$	$2n^2 + n$	$4^{\frac{1}{\frac{1}{3}n^3+7n^2+\frac{2}{3}n}}$
OCM	$\frac{2}{3}n^3 + 3n^2 - \frac{2}{3}n$	$2n^2 + n$	$4^{\frac{1}{\frac{2}{3}n^3+5n^2+\frac{1}{3}n}}$
JM	$n^3 + 5n^2 - n$	$2n^2 + 3n$	$5^{\frac{1}{n^3+7n^2+2n}}$
SM	$\frac{2}{3}n^3 + 4n^2 - \frac{2}{3}n$	$2n^2 + n$	$4^{\frac{1}{\frac{2}{3}n^3+6n^2+\frac{1}{3}n}}$
SiM	$\frac{2}{3}n^3 + 2n^2 + \frac{4}{3}n + 1$	$2n^2 + 2n$	$5^{\frac{1}{\frac{2}{3}n^3+4n^2+\frac{10}{3}n+1}}$
ACTV	$\frac{2}{3}n^3 + 5n^2 - \frac{2}{3}n$	$2n^2 + 2n$	$5^{\frac{1}{\frac{2}{3}n^3+7n^2+\frac{4}{3}n}}$
NM	$\frac{1}{3}n^3 + n^2 - \frac{1}{3}n$	$n^2 + n$	$2^{\frac{1}{\frac{1}{3}n^3+2n^2+\frac{2}{3}n}}$

Below, we compare nine schemes—GTM, GOM, CM, OCM, JM, SM, SiM, ACTV, and the NM method—over small systems ($n = 5, \ldots, 20$) and larger ones ($n = 50, \ldots, 200$). We report the computational efficiency index

$$\mathrm{CI} = p^{\frac{1}{d+\mathrm{op}}}, \qquad \ln \mathrm{CI} = \frac{\ln p}{d + \mathrm{op}},$$

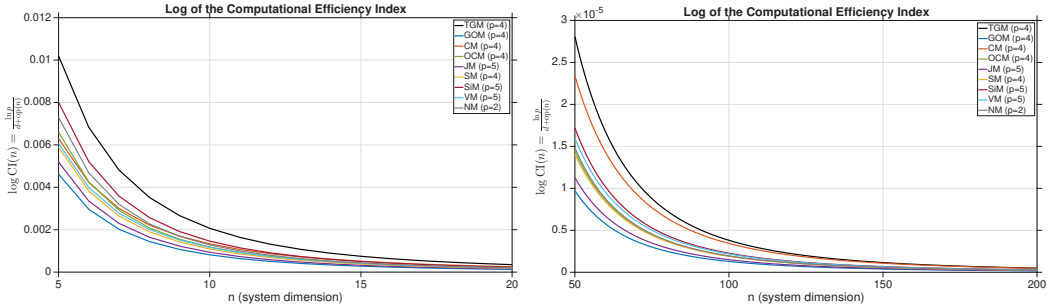
where p is the convergence order, d agregates function/derivative/divided-difference evaluations, and op the number of products-quotiens per iteration.

In Figures 1 and 2, higher curves indicate better efficiency. Figure 1 displays the values of CI and their absolute proximity to 1, whereas Figure 2 presents the same information on a logarithmic scale, which magnifies the differences among methods (while preserving the ordering induced by CI) and becomes more informative for larger values of n.



(**a**) Efficiency index for $n = 5, \ldots, 20$. (**b**) Efficiency index for $n = 50, \ldots, 200$.

Figure 1. Efficiency index $\mathrm{CI} = p^{1/(d+\mathrm{op})}$ for nine methods. Higher curves (closer to 1) indicate better efficiency; see text for discussion.



(**a**) Efficiency index for $n = 5, \ldots, 20$. (**b**) Efficiency index for $n = 50, \ldots, 200$.

Figure 2. Log of the efficiency index $\ln \mathrm{CI} = \frac{\ln p}{d+\mathrm{op}}$. Plotting $\ln \mathrm{CI}$ separates curves that cluster near 1 in Figure 1.

In Tables 3 and 4, it can be observed that all methods tend toward a computational efficiency index very close to one as the system dimension increases, which is consistent with the definition of CI and the rapid growth of the denominator $d + \mathrm{op}$ with n. However, for small and medium dimensions, clear differences emerge: TGM consistently achieves the highest efficiency values, followed by SiM and ACTV, while OCM, CM, and SM display intermediate performance. In contrast, GOM and NM present the lowest indices, confirming their reduced competitiveness under this efficiency analysis. Overall, these results indicate that TGM and SiM provide the best cost–efficiency balance for lower dimensions, whereas for higher-dimensional systems all schemes become practically indistinguishable in terms of their computational efficiency index.

Table 3. Sample values of the computational efficiency index CI for $n = 5, 10, 15, 20$, comparing the nine methods.

n	TGM	GOM	CM	OCM	JM	SM	SiM	ACTV	NM
5	1.01024547	1.00463167	1.00632123	1.00662324	1.00520524	1.00591656	1.00803930	1.00609183	1.00732297
10	1.00206815	1.00081580	1.00133386	1.00118557	1.00093616	1.00109217	1.00146287	1.00116694	1.00128443
15	1.00074720	1.00028010	1.00051168	1.00041023	1.00032323	1.00038462	1.00050292	1.00041867	1.00043741
20	1.00035182	1.00012837	1.00025301	1.00018889	1.00014848	1.00017912	1.00022991	1.00019725	1.00019920

Table 4. Sample values of the computational efficiency index CI for $n = 50, 100, 150, 200$, comparing the nine methods.

n	TGM	GOM	CM	OCM	JM	SM	SiM	ACTV	NM
50	1.00002809	1.00000973	1.00002342	1.00001446	1.00001129	1.00001410	1.00001721	1.00001595	1.00001484
100	1.00000381	1.00000130	1.00000344	1.00000193	1.00000150	1.00000191	1.00000228	1.00000218	1.00000196
150	1.00000116	1.00000039	1.00000108	1.00000059	1.00000046	1.00000058	1.00000069	1.00000067	1.00000059
200	1.00000050	1.00000017	1.00000047	1.00000025	1.00000019	1.00000025	1.00000029	1.00000029	1.00000025

4. Dynamics of Generalized Traub Methods

Consider the polynomial system

$$F(x) = \begin{pmatrix} x_1^2 - 1 \\ x_2^2 - 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \qquad x = (x_1, x_2)^T.$$

By applying the generalized Traub family to F, we obtain a *rational operator* T_γ, defined in the set $\{(x_1, x_2) \in \mathbb{R}^2 : x_1 x_2 \neq 0\}$, of the form

$$T_\gamma(x_1, x_2) = \big(t_1(x_1, x_2),\, t_2(x_1, x_2)\big), \qquad \begin{aligned} t_1(x_1, x_2) &= x_1 - \frac{\left(-6 + \frac{1}{x_1^2} + 5x_1^2\right)\Phi_\gamma(x_1, x_2)}{8\, x_1} \\ t_2(x_1, x_2) &= x_2 - \frac{\left(-6 + \frac{1}{x_2^2} + 5x_2^2\right)\Phi_\gamma(x_1, x_2)}{8\, x_2}, \end{aligned} \tag{23}$$

where

$$\Phi_\gamma(x_1, x_2) = 1 + \gamma\, \frac{(x_1^2 - 1)^4 x_2^4 + x_1^4 (x_2^2 - 1)^4}{16\, x_1^4 x_2^4 \big((x_1^2 - 1)^2 + (x_2^2 - 1)^2\big)^2}. \tag{24}$$

The Jacobian matrix associated with T_γ is given by

$$J_{T_\gamma}(x_1, x_2) = \begin{pmatrix} \partial_{x_1} t_1 & \partial_{x_2} t_1 \\ \partial_{x_1} t_2 & \partial_{x_2} t_2 \end{pmatrix}.$$

Writing

$$g(x) = \frac{-6 + \frac{1}{x^2} + 5x^2}{8x}, \qquad t_i(x_1, x_2) = x_i - g(x_i)\, \Phi_\gamma(x_1, x_2), \quad i = 1, 2,$$

and applying the product and chain rules, we obtain

$$J_{T_\gamma}(x_1, x_2) = \begin{pmatrix} 1 - g'(x_1)\Phi_\gamma(x_1, x_2) - g(x_1)\Phi_{\gamma, x_1}(x_1, x_2) & -g(x_1)\Phi_{\gamma, x_2}(x_1, x_2) \\ -g(x_2)\Phi_{\gamma, x_1}(x_1, x_2) & 1 - g'(x_2)\Phi_\gamma(x_1, x_2) - g(x_2)\Phi_{\gamma, x_2}(x_1, x_2) \end{pmatrix}, \tag{25}$$

where $\Phi_{\gamma, x_i} = \partial_{x_i} \Phi_\gamma$, $i = 1, 2$.

The set of roots of F, namely $\mathcal{C}_0 = \{(\pm 1, \pm 1)\}$, are not fixed points of the full operator T_γ, because the coupling is not defined at those points. To analyze the behavior when both components evolve symmetrically, we restrict the dynamics to the diagonal by imposing

$$x_1^2 = x_2^2 \quad \Longrightarrow \quad x_1^4 = x_2^4,$$

which yields a simplified one-dimensional operator

$$T_\gamma^{\mathrm{red}}(x_1, x_2) = \big(g_\gamma(x_1),\, g_\gamma(x_2)\big), \qquad g_\gamma(x_i) = x_i - \frac{\big(-6 + \frac{1}{x_i^2} + 5x_i^2\big)\big(1 + \Phi_\gamma^{\mathrm{red}}(x_i)\big)}{8\, x_i}, i = 1, 2, \tag{26}$$

and the reduced coupling becomes

$$\Phi_\gamma^{\mathrm{red}}(x_i) = 1 + \frac{\gamma}{16}\,\frac{(x_i^2 - 1)^2}{x_i^4}.$$

Within this reduced dynamic, the points $(\pm 1, \pm 1)$ act as genuine fixed points (and, as will be shown later, they are superattracting). The following result summarizes the complete characterization of fixed points and their nature.

Theorem 4. *Regarding the operators T_γ and T_γ^{red}, the following findings were obtained.*

(*i*) *For the reduced operator T_γ^{red} defined in (26), the set*

$$C_0 = \{(\pm 1, \pm 1)\}$$

consists of fixed points, and each of its elements exhibits a superattracting behavior.

(*ii*) *For the operator T_γ defined in (23), the points $(\pm s, \pm s)$, with $s = 1/\sqrt{5}$, are fixed points. At $x_1 = x_2 = \pm s$, the Jacobian has identical eigenvalues*

$$\lambda_1 = \lambda_2 = 1 + 5\,\Phi_\gamma(s, s) = 6 + \frac{125}{32}\,\gamma.$$

These points are attractors when $\gamma \in \left(-\frac{224}{125}, -\frac{160}{125}\right)$.

(*iii*) *For the operator T_γ defined in (23), the mixed points $(\pm 1, \pm s)$ and $(\pm s, \pm 1)$, with $s = 1/\sqrt{5}$, are fixed points. At these points, the Jacobian matrix is diagonal (25), with*

$$\lambda_1 = 1 - \Phi_\gamma(1, s), \qquad \lambda_2 = 1 + 5\,\Phi_\gamma(1, s), \qquad \Phi_\gamma(1, s) = 1 + \frac{25}{16}\gamma.$$

Since the stability conditions for λ_1 and λ_2 cannot be satisfied simultaneously, these points are never attractors. More precisely,

$$\begin{cases} \text{they are saddle points} & \text{if } \gamma \in \left(-\frac{112}{125}, \frac{16}{25}\right), \\ \text{they are repeller points} & \text{if } \gamma < -\frac{112}{125} \text{ or } \gamma > \frac{16}{25}, \end{cases}$$

and at the threshold values

$$\gamma \in \left\{-\frac{112}{125}, \frac{16}{25}\right\},$$

they are neutral (parabolic).

Proof. The proof proceeds by first establishing part (i). For the reduced operator

$$T_\gamma^{\mathrm{red}}(x_1, x_2) = \big(g_\gamma(x_1),\, g_\gamma(x_2)\big),$$

where

$$g_\gamma(x_i) = x_i - \frac{A(x_i)\,\Phi_\gamma^{\mathrm{red}}(x_i)}{8x_i}, \qquad A(x_i) = 5x_i^2 + \frac{1}{x_i^2} - 6,$$

and

$$\Phi_\gamma^{\mathrm{red}}(x_i) = 1 + \frac{\gamma}{16}\frac{(x_i^2-1)^2}{x_i^4}, i = 1,2,$$

we analyze the behavior at the points $x_i = \pm 1$.

Observe that

$$A(\pm 1) = 5(1) + 1 - 6 = 0, \qquad \Phi_\gamma^{\mathrm{red}}(\pm 1) = 1.$$

Hence,

$$g_\gamma(\pm 1) = \pm 1,$$

so that the points $(\pm 1, \pm 1)$ are fixed points of T_γ^{red}.

To determine their nature, we compute the derivative:

$$g_\gamma'(x_i) = 1 - \frac{A'(x_i)\Phi_\gamma^{\mathrm{red}}(x_i) + A(x_i)(\Phi_\gamma^{\mathrm{red}})'(x_i)}{8x_i} + \frac{A(x_i)\Phi_\gamma^{\mathrm{red}}(x_i)}{8x_i^2}.$$

Since $A(\pm 1) = 0$, all terms containing $A(x_i)$ vanish at $x_i = \pm 1$, and therefore

$$g_\gamma'(\pm 1) = 1 - \frac{A'(\pm 1)}{8(\pm 1)}.$$

Now,

$$A'(x_i) = 10x_i - \frac{2}{x_i^3},$$

so that $|A'(\pm 1)| > 1$. Since the reduced operator is uncoupled,

$$JT^{\mathrm{red}}{}_\gamma(x_1, x_2) = \begin{pmatrix} g_\gamma'(x_1) & 0 \\ 0 & g_\gamma'(x_2) \end{pmatrix}.$$

Evaluating at $(\pm 1, \pm 1)$ and using $g_\gamma'(\pm 1) = 0$, we obtain

$$JT^{\mathrm{red}}{}_\gamma(\pm 1, \pm 1) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix},$$

and thus each of the four points $(\pm 1, \pm 1)$ is superattracting.

We next establish part (ii).

Let $s = 1/\sqrt{5}$. From (23), write $T_\gamma = (t_1, t_2)$ with

$$t_i(x_1, x_2) = x_i - g(x_i)\,\Phi_\gamma(x_1, x_2), \qquad i \in \{1, 2\},$$

where

$$g(x_i) = \frac{-6 + \frac{1}{x_i^2} + 5x_i^2}{8x_i}.$$

If (x_1, x_2) is a fixed point of T_γ, then $t_i(x_1, x_2) = x_i$, hence

$$g(x_i)\,\Phi_\gamma(x_1, x_2) = 0, \qquad i = 1, 2.$$

In particular, any point satisfying $g(x_1) = g(x_2) = 0$ is a fixed point. Now,

$$g(x_i) = 0 \iff -6 + \frac{1}{x_i^2} + 5x_i^2 = 0 \iff 5x_i^4 - 6x_i^2 + 1 = 0 \iff (5x_i^2 - 1)(x_i^2 - 1) = 0.$$

Therefore, $x_i \in \{\pm 1, \pm s\}$, and hence $(\pm s, \pm s), (\pm 1, \pm s), (\pm s, \pm 1)$ are fixed points of T_γ. To determine their local behavior, we evaluate the Jacobian at these points. From (25), the Jacobian matrix of T_γ is

$$J_{T_\gamma}(x_1, x_2) = \begin{pmatrix} 1 - g'(x_1)\Phi_\gamma - g(x_1)\Phi_{\gamma,x_1} & -g(x_1)\Phi_{\gamma,x_2} \\ -g(x_2)\Phi_{\gamma,x_1} & 1 - g'(x_2)\Phi_\gamma - g(x_2)\Phi_{\gamma,x_2} \end{pmatrix},$$

where $\Phi_{\gamma,x_i} = \partial_{x_i}\Phi_\gamma$. At $(x_1, x_2) = (\pm s, \pm s)$, we have $g(\pm s) = 0$, so the off-diagonal entries vanish and the Jacobian becomes diagonal:

$$J_{T_\gamma}(\pm s, \pm s) = \begin{pmatrix} 1 - g'(\pm s)\Phi_\gamma(s, s) & 0 \\ 0 & 1 - g'(\pm s)\Phi_\gamma(s, s) \end{pmatrix}.$$

Hence, the eigenvalues are identical and satisfy

$$\lambda_1 = \lambda_2 = 1 - g'(\pm s)\,\Phi_\gamma(s, s).$$

To compute $g'(\pm s)$, note that

$$g(x_i) = \frac{1}{8}\left(-\frac{6}{x_i} + \frac{1}{x_i^3} + 5x_i\right) \quad \Longrightarrow \quad g'(x_i) = \frac{1}{8}\left(5 + \frac{6}{x_i^2} - \frac{3}{x_i^4}\right) \qquad i = 1, 2.$$

Since $s^2 = 1/5$, we have $1/s^2 = 5$ and $1/s^4 = 25$, and thus

$$g'(s) = \frac{1}{8}(5 + 30 - 75) = -5, \qquad g'(-s) = -5.$$

Consequently,

$$\lambda_1 = \lambda_2 = 1 - (-5)\Phi_\gamma(s, s) = 1 + 5\,\Phi_\gamma(s, s).$$

Finally, evaluating Φ_γ at (s, s) gives

$$\Phi_\gamma(s, s) = 1 + \frac{25}{32}\gamma,$$

and therefore

$$\lambda_1 = \lambda_2 = 1 + 5\left(1 + \frac{25}{32}\gamma\right) = 6 + \frac{125}{32}\gamma.$$

Thus, the fixed points $(\pm s, \pm s)$ are genuine attractors precisely when $|\lambda_1| < 1$ (see Theorem 1), that is,

$$-1 < 6 + \frac{125}{32}\gamma < 1 \iff \gamma \in \left(-\frac{224}{125}, -\frac{160}{125}\right).$$

Finally, we prove part (iii). From (25),

$$J_{T_\gamma}(x_1, x_2) = \begin{pmatrix} 1 - g'(x_1)\Phi_\gamma - g(x_1)\Phi_{\gamma,x_1} & -g(x_1)\Phi_{\gamma,x_2} \\ -g(x_2)\Phi_{\gamma,x_1} & 1 - g'(x_2)\Phi_\gamma - g(x_2)\Phi_{\gamma,x_2} \end{pmatrix}, \qquad \Phi_{\gamma,x_i} = \partial_{x_i}\Phi_\gamma.$$

At any point where $g(x_1) = g(x_2) = 0$ the off-diagonal entries vanish and the Jacobian becomes diagonal, so its eigenvalues are the diagonal entries. For mixed fixed points, say $(x_1, x_2) = (\pm 1, \pm s)$, we obtain

$$\lambda_1 = 1 - g'(\pm 1)\,\Phi_\gamma(1,s), \qquad \lambda_2 = 1 - g'(\pm s)\,\Phi_\gamma(1,s),$$

(where we used that $\Phi_\gamma(\pm 1, \pm s) = \Phi_\gamma(1,s)$ since Φ_γ depends on x_1^2, x_2^2 only (24)). Now

$$g'(x_i) = \frac{1}{8}\left(5 + \frac{6}{x_i^2} - \frac{3}{x_i^4}\right), \qquad i = 1,2,$$

hence, $g'(\pm 1) = 1$ and $g'(\pm s) = -5$. Therefore,

$$\lambda_1 = 1 - \Phi_\gamma(1,s), \qquad \lambda_2 = 1 + 5\,\Phi_\gamma(1,s).$$

From the explicit formula of Φ_γ, using $s^2 = 1/5$ and $s^4 = 1/25$, we compute

$$\Phi_\gamma(1,s) = 1 + \gamma\,\frac{(1^2-1)^4 s^4 + 1^4 (s^2-1)^4}{16 \cdot 1^4 s^4 \big((1^2-1)^2 + (s^2-1)^2\big)^2} = 1 + \gamma\,\frac{(s^2-1)^4}{16 s^4 (s^2-1)^4} = 1 + \frac{25}{16}\gamma.$$

Let $\Phi := \Phi_\gamma(1,s) = 1 + \frac{25}{16}\gamma$. The mixed fixed points are attracting only if $|\lambda_1| < 1$ and $|\lambda_2| < 1$ hold simultaneously. However,

$$|\lambda_1| < 1 \iff |1 - \Phi| < 1 \iff 0 < \Phi < 2,$$

whereas

$$|\lambda_2| < 1 \iff |1 + 5\Phi| < 1 \iff -\frac{2}{5} < \Phi < 0.$$

These conditions are incompatible; hence, the mixed fixed points are never attractors.

More precisely, the mixed points are saddles when one eigenvalue has modulus less than 1 and the other has modulus greater than 1, which occurs for

$$-\frac{2}{5} < \Phi < 2 \iff -\frac{112}{125} < \gamma < \frac{16}{25}.$$

They are repellers when both eigenvalues satisfy $|\lambda_1| > 1$ and $|\lambda_2| > 1$, which happens for $\Phi < -\frac{2}{5}$ or $\Phi > 2$, i.e.,

$$\gamma < -\frac{112}{125} \quad \text{or} \quad \gamma > \frac{16}{25}.$$

At the threshold values $\gamma \in \left\{-\frac{112}{125}, \frac{16}{25}\right\}$, one eigenvalue lies on the unit circle (in fact, $\lambda = -1$), so these fixed points are neutral. □

Remark 1. *We introduce the auxiliary functions*

$$A(x_i) = 5x_i^2 + \frac{1}{x_i^2} - 6, \qquad i = 1,2,$$

$$\Phi_\gamma(x_1,x_2) = 1 + \gamma\,\mathcal{M}(x_1,x_2),$$

where

$$\mathcal{M}(x_1,x_2) = \frac{(x_1^2-1)^4 x_2^4 + x_1^4 (x_2^2-1)^4}{16\,x_1^4 x_2^4 \big((x_1^2-1)^2 + (x_2^2-1)^2\big)^2}.$$

We consider the domain

$$\mathcal{D} = \Big\{(x_1,x_2) \in \mathbb{R}^2 :\ x_1 x_2 \neq 0,\ \big((x_1^2-1)^2 + (x_2^2-1)^2\big) \neq 0\Big\}.$$

The fixed-point equation $T_\gamma(x_1, x_2) = (x_1, x_2)$ is equivalent to

$$A(x_i)\,\Phi_\gamma(x_1, x_2) = 0, \qquad i = 1, 2.$$

Besides the solutions given by $A(x_i) = 0$, one must also consider the possibility

$$\Phi_\gamma(x_1, x_2) = 0.$$

This gives rise to the set

$$\Omega_\gamma = \{(x_1, x_2) \in \mathcal{D} : \ \Phi_\gamma(x_1, x_2) = 0\}.$$

Observe that $\mathcal{M} \geq 0$ on $\mathcal{D}$ and, in fact, $\mathcal{M} > 0$ throughout $\mathcal{D}$. Hence

$$\Phi_\gamma(x_1, x_2) = 1 + \gamma\mathcal{M}(x_1, x_2).$$

If $\gamma \geq 0$, then $\Phi_\gamma(x_1, x_2) > 0$ on $\mathcal{D}$, so $\Omega_\gamma = \varnothing$.
Assume now that $\gamma < 0$ and write $c := -1/\gamma > 0$. Then $\Phi_\gamma(x_1, x_2) = 0$ is equivalent to

$$\mathcal{M}(x_1, x_2) = c.$$

Since $\mathcal{M}$ is continuous on each connected component of $\mathcal{D}$ and becomes unbounded when approaching the excluded points $(\pm 1, \pm 1)$ from within $\mathcal{D}$, it follows that for every $c > 0$ there exists $x \in \mathcal{D}$ such that $\mathcal{M}(x_1, x_2) = c$.

Therefore, when $\gamma < 0$, the set Ω_γ is nonempty. This set forms a non-isolated continuum of stationary states along which the iteration remains constant, producing stagnation of the method.

Dynamical Planes

To illustrate the basins of attraction generated by the Generalized Traub family when applied to the system

$$F(x) = (x_1^2 - 1,\ x_2^2 - 1)^T = (0, 0)^T,$$

we compute dynamical planes on a prescribed rectangular region, discretized through a uniform grid. From each grid point used as a seed, the iterative operator is applied repeatedly until convergence is detected (or until a prescribed maximum number of iterations is reached), and the node is then colored according to its limiting behavior.

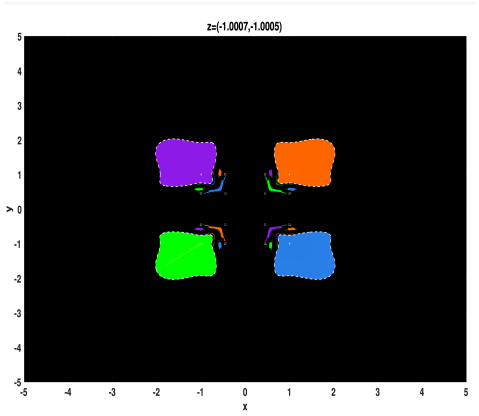
By default, iterations are carried out using the operator (23). However, whenever the starting point satisfies $x_1^2 \approx x_2^2$ (within a small tolerance), the reduced operator (26) is used instead, in order to accurately reflect the dynamics along the diagonal or antidiagonal directions of the phase space.

The visualization follows the scheme: a fixed palette of blue, orange, green, and purple is associated with the basins leading to the four roots $(\pm 1, \pm 1)$, which are also indicated in the figures by star markers. Basins that converge to extraneous fixed points (whenever these behave as attractors) are depicted in gold, and such points are identified with circles. Finally, the nodes that fail to converge to any fixed point—either due to divergence or because they reach the iteration limit without stabilization—are colored in black.

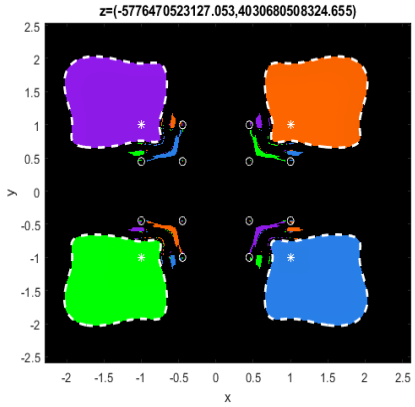
In Figure 3 ($\gamma < 0$), a continuous curve of points Ω_γ emerges, which organizes the dynamics and delimits the transitions between basins. For $\gamma = -1.4$ (Figure 4a and detail Figure 4b), basins associated with strange attractors $(\pm s, \pm s)$, with $s = 1/\sqrt{5}$, appear in the interval $\gamma \in \left(-\frac{224}{125}, -\frac{160}{125}\right)$. As γ decreases, these attractors vanish: for $\gamma = -5$ (Figure 3c and detail Figure 3d), convergence is restricted to $(\pm 1, \pm 1)$, while for $\gamma = -30$ (Figure 3a and detail Figure 3b), the curve Ω_γ deforms, partially covering the basins, resulting in a

noticeable enlargement of the black regions in the dynamical portrait, suggestive of slow convergence or possible divergence.

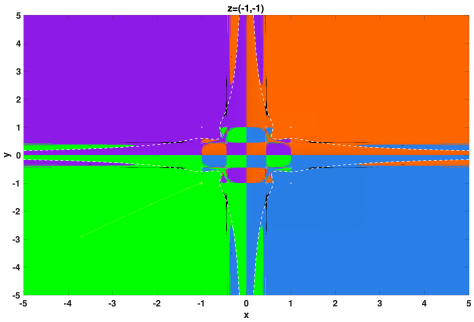
Figure 5 ($\gamma \geq 0$) shows the disappearance of strange attractors. For $\gamma = 0$ (Figure 5a), the rational operator associated with the classical Traub method for nonlinear systems is obtained, with convergence order three, yielding wide and regular basins that converge exclusively to the roots $(\pm 1, \pm 1)$. In Figure 5b, with $\gamma = 2$, we illustrate the dynamical behavior of the rational operator associated with the unique optimal method in the Generalized Traub family according to the Cordero–Torregrosa criterion; in this case, a slight intertwining of the basin boundaries appears. This interlacing becomes more pronounced for $\gamma = 10$ (Figure 5c), where several non-immediate connected components of the attraction basins become prominent, reflecting the increasing fragmentation of the dynamical plane. Finally, for $\gamma = 40$ (Figure 5d), a pronounced fractal-like structure emerges, with heavily intertwined basins and dominant non-convergent regions, associated with divergence or iteration cutoff. Nevertheless, whenever the iteration converges, it invariably does so towards one of the four system roots.



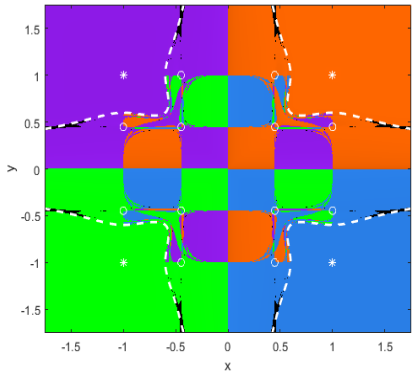
(**a**) $\gamma = -30$.



(**b**) $\gamma = -30$, zoom.



(**c**) $\gamma = -5$.

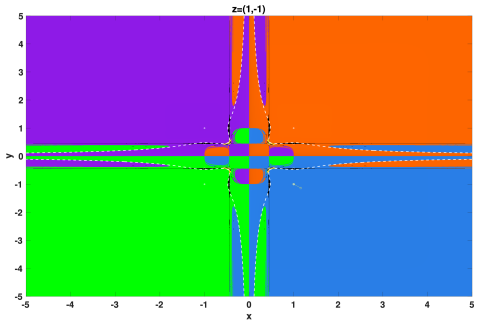


(**d**) $\gamma = -5$, zoom.

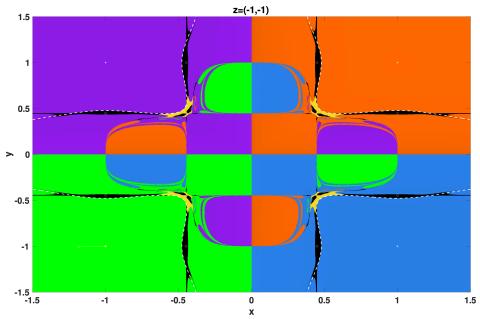
Figure 3. Dynamical portraits corresponding to several negative choices of the parameter γ.

To conclude this dynamical study, we emphasize that the parameter γ plays a decisive role in the practical robustness of the Generalized Traub family. For negative values of γ, the emergence of the continuous curve Ω_γ and, in some ranges, of strange attractors $(\pm s, \pm s)$, leads to a deterioration of the convergence domains, with larger non-convergent regions and possible stagnation or divergence. By contrast, for nonnegative values of γ, the strange attractors disappear and the method converges only to the actual roots of the system, although the geometry of the basin boundaries becomes increasingly intricate as γ grows. In particular, the case $\gamma = 2$ is especially relevant from both the theoretical and

practical viewpoints: besides corresponding to the unique optimal fourth-order member of the family under the Cordero–Torregrosa criterion, it also preserves a reliable global behavior, with convergence restricted to the true roots and without the undesirable attracting structures observed for negative values of γ. These dynamical findings are consistent with the numerical experiments presented in the next section, where the method with $\gamma = 2$ exhibits a favorable balance between convergence order, efficiency, and stability.

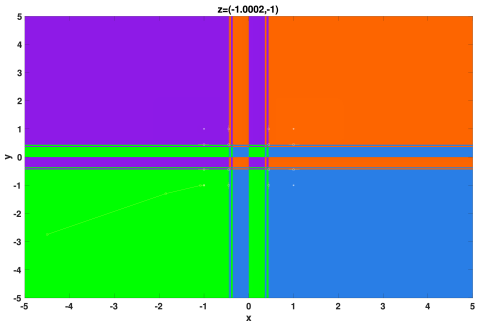


(**a**) $\gamma = -1.4$.

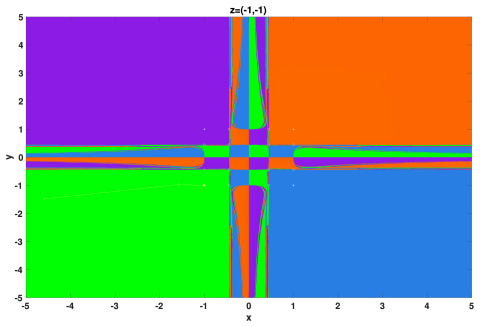


(**b**) $\gamma = -1.4$, zoom.

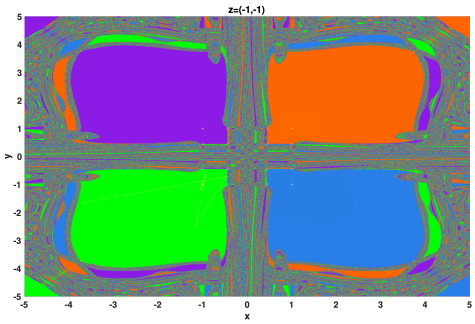
Figure 4. Negative parameter γ.



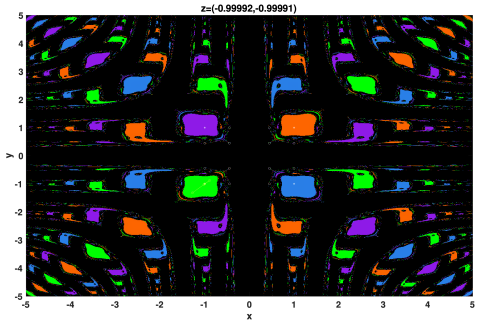
(**a**) $\gamma = 0$.



(**b**) $\gamma = 2$.



(**c**) $\gamma = 10$.



(**d**) $\gamma = 40$.

Figure 5. Dynamical portraits corresponding to several positive choices of the parameter γ.

5. Numerical Experiments

This section describes the computational framework adopted to compare the performance of the iterative schemes under a unified and controlled environment. For each method, the reported tables include the total number of iterations required to satisfy the stopping criterion, the norm $\left\|x^{(k+1)} - x^{(k)}\right\|$ measuring the distance between two last iterations, and the residual norm $\left\|F\left(x^{(k+1)}\right)\right\|$ evaluated at the final iterate. The execution time in seconds (e-time) is also recorded to quantify the computational cost, and the estimated

asymptotic computational order of convergence (ACOC), is provided in order to assess the local convergence behavior of each scheme.

To ensure a fair and consistent comparison, a common computational configuration was employed in all experiments. The stopping condition was defined as

$$\left\|x^{(k+1)} - x^{(k)}\right\| + \left\|F(x^{(k+1)})\right\| < 10^{-100},$$

with a maximum of 50 iterations permitted, whichever occurred first. All computations were performed using variable precision arithmetic via `vpa` with 1000 digits of precision. This high-precision setting effectively suppresses round-off errors and allows the asymptotic convergence regime of each method to be accurately observed.

To obtain reliable timing measurements, each method was executed 15 times for every test problem. The execution time reported in the tables corresponds to the average of these runs. This averaging procedure mitigates variability caused by background processes and system-level fluctuations, thereby providing a stable and representative estimate of the computational effort. Moreover, all schemes were tested using the same initial approximation for each nonlinear system, guaranteeing methodological uniformity across the comparisons.

The numerical experiments were conducted on a computer running macOS Tahoe (version 26.3), equipped with an Apple M2 processor and 8 GB of LPDDR5 RAM (model year 2023). These specifications ensure that the reported execution times correspond to a modern mid-range computational environment.

The nonlinear systems considered in the numerical experiments are described below. For each system, we present its analytical formulation together with the corresponding Jacobian matrix, since both components are required for the implementation of the iterative procedures.

Example 1. *Consider the following nonlinear system composed of m equations:*

$$f_i(x) = x_i - 1.5 \sin\left(\sum_{\substack{j=1 \\ j\neq i}}^{m} x_j\right) = 0, \qquad i = 1, \ldots, m.$$

For the numerical experiment, we set $m = 20$*. As initial approximation, we take*

$$x^{(0)} = \left(\tfrac{1}{10}, \tfrac{1}{10}, \ldots, \tfrac{1}{10}\right)^T,$$

and the iterative schemes converge to the approximate solution

$$x^* \approx (0.15973, \ldots, 0.15973)^T.$$

Example 2. *We study the nonlinear system, for* $n = 100$*, defined component-wise by*

$$f_i(x) = 4x_i^2 + 1 + \arctan(x_i) - 2\sum_{j=1}^{n} x_j^2, \qquad i = 1, \ldots, n.$$

For the numerical test, the starting vector is selected as

$$x^{(0)} = \left(\tfrac{1}{10}, \tfrac{1}{10}, \ldots, \tfrac{1}{10}\right)^T,$$

and the iterations converge to the approximate solution

$$x^* \approx (0.07402032, \ldots, 0.07402032)^T.$$

Example 3. *We consider the nonlinear system composed of m equations*

$$f_i(x) = x_i - \cos\left(-\sum_{j=1}^{m} x_j + 2x_i\right) = 0, \qquad i = 1, \ldots, m,$$

with $m = 100$.

The initial approximation is chosen as

$$x^{(0)} = \left(\tfrac{1}{10}, \tfrac{1}{10}, \ldots, \tfrac{1}{10}\right)^T.$$

Several iterative schemes converge to the symmetric solution

$$x^* = (0.113358953445, \ldots, 0.113358953445)^T.$$

Due to the periodic nature of the cosine function, different methods may converge to distinct solutions depending on the basin of attraction and the dynamical structure of the scheme.

The performance metrics (number of iterations, residual norm, computational time, and ACOC) are reported in Table 7.

Remark 2. *It is worth observing that the methods of GTM, GOM, CM, OCM, SM, and JM converge to the same approximate solution*

$$x^* = 0.113358953445\ldots,$$

whereas SiM and NM converge to

$$x^* = 0.2429319603\ldots,$$

and the ACTV method converges to

$$x^* = 0.142794718805\ldots.$$

This behavior is typical in nonlinear systems involving trigonometric functions, since the cosine function is periodic and therefore admits many roots. Consequently, different iterative schemes may converge to distinct solutions depending on their dynamical structure and the basin of attraction associated with the chosen initial approximation.

Example 4. *We study the nonlinear system defined by*

$$f_i(x) = x_i + \log(2 + x_i + x_{i+1}) = 0, \qquad i = 1, \ldots, 999,$$

together with the closing equation

$$f_{1000}(x) = x_{1000} + \log(2 + x_{1000} + x_1) = 0.$$

The starting vector for the numerical experiments is

$$x^{(0)} = \left(\tfrac{1}{10}, \tfrac{1}{10}, \ldots, \tfrac{1}{10}\right)^T,$$

and the computed symmetric solution is

$$x^* = (-0.314923, \ldots, -0.314923)^T.$$

General Discussion of the Numerical Results

From the numerical results presented in Tables 5–8, several relevant conclusions can be drawn regarding computational efficiency, stability, and robustness.

Table 5. Numerical results for Example 1.

Method	Iter	$\|x^{(k+1)} - x^{(k)}\|$	$\|F(x^{(k+1)})\|$	e-Time	ACOC
GTM	6	1.8694×10^{-366}	$1.22054 \times 10^{-1826}$	0.778765	5.0
GOM	5	2.39383×10^{-115}	6.20273×10^{-458}	9.006553	4.0
CM	6	3.79955×10^{-234}	2.12822×10^{-932}	10.464925	4.0
OCM	6	2.93871×10^{-250}	1.42037×10^{-997}	10.123275	4.0
SM	7	7.28143×10^{-268}	$4.82567 \times 10^{-1007}$	1.691032	4.0
JM	5	1.43332×10^{-160}	3.18463×10^{-799}	1.061132	5.0
SiM	5	4.22607×10^{-155}	1.14672×10^{-771}	0.607800	5.0
ACTV	50	7.33782×10^{-2}	1.03868×10^{1}	7.748143	–
NM	50	2.42758×10^{1}	1.09698×10^{2}	3.169261	–

Table 6. Numerical results for Example 2.

Method	Iter	$\|x^{(k+1)} - x^{(k)}\|$	$\|F(x^{(k+1)})\|$	e-Time	ACOC
GTM	5	1.69088×10^{-174}	2.35403×10^{-694}	19.877017	4.0
GOM	5	1.98891×10^{-244}	1.50256×10^{-974}	216.137642	4.0
CM	5	2.86706×10^{-161}	3.24320×10^{-641}	170.043090	4.0
OCM	5	1.06741×10^{-208}	1.24596×10^{-831}	130.544407	4.0
SM	5	2.62381×10^{-183}	1.06155×10^{-729}	97.805743	4.0
JM	5	3.33484×10^{-489}	$1.52688 \times 10^{-1007}$	45.761464	5.0
SiM	5	1.79100×10^{-433}	$2.42251 \times 10^{-1007}$	20.220612	5.0
ACTV	5	3.48429×10^{-487}	$2.38910 \times 10^{-1007}$	32.253352	5.0
NM	8	1.24737×10^{-104}	3.05075×10^{-207}	17.608970	2.0

Table 7. Numerical results for Example 3.

Method	Iter	$\|x^{(k+1)} - x^{(k)}\|$	$\|F(x^{(k+1)})\|$	e-Time	ACOC
GTM	5	1.07581×10^{-113}	1.24903×10^{-449}	22.083587	4.0
GOM	6	3.9684×10^{-206}	2.20488×10^{-819}	1460.736612	4.0
CM	6	1.80757×10^{-303}	$7.07965 \times 10^{-1007}$	1733.671570	4.0
OCM	6	1.03973×10^{-156}	1.04915×10^{-621}	3286.421179	4.0
SM	7	2.44114×10^{-136}	3.09833×10^{-540}	218.556001	4.0
JM	5	2.46773×10^{-150}	5.85396×10^{-746}	79.156905	5.0
SiM	5	1.01276×10^{-174}	5.93545×10^{-867}	46.264154	5.0
ACTV	6	2.46896×10^{-347}	$4.74734 \times 10^{-1007}$	70.370451	5.0
NM	12	5.42331×10^{-181}	3.43112×10^{-359}	58.118465	2.0

Table 8. Numerical results for Example 4.

Method	Iter	$\|x^{(k+1)} - x^{(k)}\|$	$\|F(x^{(k+1)})\|$	e-Time	ACOC
GTM	5	7.85761×10^{-256}	0.0	67.317130	4.0
GOM	–	–	–	–	–
CM	–	–	–	–	–
OCM	–	–	–	–	–
SM	5	2.20215×10^{-231}	1.06649×10^{-928}	743.735748	4.0
JM	4	3.96175×10^{-109}	6.93616×10^{-550}	108.908601	5.0
SiM	4	9.57416×10^{-105}	3.43675×10^{-528}	54.457085	5.0
ACTV	4	9.36462×10^{-107}	4.65544×10^{-538}	73.733254	5.0
NM	8	2.37655×10^{-104}	1.90276×10^{-209}	53.762671	2.0

First, in Example 1, NM fails to converge within the prescribed number of iterations, while the proposed GTM converges in only six iterations with a remarkably small execution time (0.778765 s) and extremely small residual norm. In contrast, classical fourth-order methods such as CM and OCM require significantly larger computational times (above 10 s), despite achieving the same theoretical order. This clearly indicates that the proposed scheme provides a better balance between convergence speed and computational cost.

In Example 2, where the dimension increases to $n = 100$, the efficiency advantage becomes more evident. Although NM converges in eight iterations, it produces residuals that are several orders of magnitude larger than those obtained by the high-order schemes. GTM converges in five iterations with competitive computational time (19.877017 s), significantly outperforming classical fourth-order methods such as GOM and CM, whose execution times grow dramatically (exceeding 130–200 s). Notably, the proposed method achieves results comparable to fifth-order methods such as JM and SiM, but with reduced computational overhead.

Example 3 further confirms this behavior. While NM requires 12 iterations and a computational time of 58.118465 s, GTM converges in only 5 iterations with a smaller execution time (22.083587 s) and much higher precision. Moreover, several classical schemes exhibit extremely large execution times (above 1400 s), highlighting scalability limitations in higher-dimensional trigonometric systems.

The most significant evidence appears in Example 4, where the dimension reaches 1000. In this large-scale problem, NM converges in eight iterations with an execution time of 53.762671 s, whereas GTM converges in five iterations with comparable computational time (67.317130 s) but substantially higher precision. Furthermore, several traditional methods (GOM, CM, and OCM) fail to converge or are not computationally feasible under the same stopping criteria, revealing structural stability issues in high-dimensional settings.

Overall, the proposed GTM consistently shows:

- A reduced number of iterations compared to NM;
- Higher precision residuals;
- Competitive or significantly lower execution times compared to classical fourth-order schemes (GOM, CM, OCM, SM);
- Greater robustness in high-dimensional nonlinear systems.

Although fifth-order methods such as JM and SiM remain highly competitive, the proposed approach achieves similar performance with lower structural complexity, reinforcing its practical efficiency.

Finally, in all experiments, the estimated ACOC values coincide with the theoretical orders of convergence, except in Table 7, where the estimated value is affected by the conditioning of the problem and the strict stopping criteria rather than by any degradation of the method itself. In well-conditioned systems, the observed convergence orders remain fully consistent with the theoretical analysis.

As a natural continuation of this work, one of the main directions for future research is the construction of new optimal iterative methods of higher order for nonlinear systems. In particular, the strategy developed in this paper may serve as a basis for deriving richer parametric families that increase the convergence order while maintaining the efficiency requirements imposed by the Cordero–Torregrosa conjecture.

Author Contributions: Conceptualization, A.C. and M.A.L.S.; methodology, M.A.L.S. and J.R.T.; software, M.A.L.S. and A.R.C.; validation, M.P.V. and J.R.T.; formal analysis, M.A.L.S. and A.R.C.; investigation, A.C., M.A.L.S., J.R.T. and A.R.C.; visualization, A.R.C.; writing—original draft preparation, M.A.L.S.; writing—review and editing, A.C., J.R.T., M.P.V. and A.R.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The MATLAB R2025b codes used in the numerical experiments are available from the corresponding author upon reasonable request.

Acknowledgments: The authors gratefully acknowledge the institutional support received from ISFODOSU, UNAPEC, UASD, INTEC, and UPV, which facilitated the development of this research. In particular, the authors thank Universidad Autónoma de Santo Domingo (UASD) for supporting an internal research project awarded to A.R.C., which contributed to the real dynamical analysis of the proposed method. The authors sincerely thank the reviewers for their careful evaluation of our manuscript and for their valuable comments and suggestions. We believe that their observations were very insightful and have helped us to substantially improve the quality, clarity, and presentation of the paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Arroyo, V.; Cordero, A.; Torregrosa, J.R. Approximation of Artificial Satellites' Preliminary Orbits: The Efficiency Challenge. *Math. Comput. Model.* **2011**, *54*, 1802–1807. [CrossRef]
2. Singh, H.; Sharma, J.R.; Kumar, S. A Simple Yet Efficient Two-Step Fifth-Order Weighted-Newton Method for Nonlinear Models. *Numer. Algorithms* **2023**, *93*, 203–225. [CrossRef]
3. Cordero, A.; Rojas-Hiciano, R.V.; Torregrosa, J.R.; Vassileva, M.P. A Highly Efficient Class of Optimal Fourth-Order Methods for Solving Nonlinear Systems. *Numer. Algorithms* **2024**, *95*, 1879–1904. [CrossRef]
4. Ostrowski, A.M. *Solution of Equations in Euclidean and Banach Spaces*; Academic Press: New York, NY, USA, 1973.
5. King, R.F. A Family of Fourth Order Methods for Nonlinear Equations. *SIAM J. Numer. Anal.* **1973**, *10*, 876–879. [CrossRef]
6. Chun, C.; Lee, M.Y.; Neta, B.; Džunić, J. On Optimal Fourth-Order Iterative Methods Free from Second Derivative and Their Dynamics. *Appl. Math. Comput.* **2012**, *218*, 6427–6438. [CrossRef]
7. Cordero, A.; Torregrosa, J.R.; Triguero-Navarro, P. First optimal vectorial eighth-order iterative scheme for solving non-linear systems. *Appl. Math. Comput.* **2025**, *498*, 129401. [CrossRef]
8. Budzko, D.; Cordero, A.; Torregrosa, J.R. A New Family of Iterative Methods Widening Areas of Convergence. *Appl. Math. Comput.* **2015**, *252*, 405–417. [CrossRef]
9. Khirallah, M.Q.; Hafiz, M.A. Solving System of Non-Linear Equations Using Family of Jarratt Methods. *Int. J. Differ. Equ. Appl.* **2013**, *12*, 69–83.
10. Chen, F.; Tao, H.; Zhuang, Z.; Paszke, W.; Stojanovic, V. Iterative learning control optimization strategy for feedback control systems with varying tasks. *Math. Model. Control* **2025**, *5*, 321. [CrossRef]
11. Zhao, J.; Zhao, H.; Song, Y.; Sun, Z.Y.; Yu, D. Fast finite-time consensus protocol for high-order nonlinear multi-agent systems based on event-triggered communication scheme. *Appl. Math. Comput.* **2026**, *508*, 129631. [CrossRef]
12. Traub, J.F. *Iterative Methods for the Solution of Equations*; American Mathematical Society: Providence, RI, USA, 1982.
13. Cordero, A.; Hueso, J.L.; Martínez, E.; Torregrosa, J.R. A Modified Newton–Jarratt Composition. *Numer. Algorithms* **2010**, *55*, 87–99. [CrossRef]
14. Amat, S.; Busquier, S.; Plaza, S. Review of Some Iterative Root-Finding Methods from a Dynamical Point of View. *Sci. Ser. A Math. Sci.* **2004**, *10*, 3–35.
15. Geum, Y.H.; Kim, Y.I.; Neta, B. A Sixth-Order Family of Three-Point Modified Newton-Like Multiple-Root Finders and the Dynamics Behind Their Extraneous Fixed Points. *Appl. Math. Comput.* **2016**, *283*, 120–140. [CrossRef]
16. Devaney, R. *An Introduction to Chaotic Dynamical Systems*; CRC Press: Boca Raton, FL, USA, 2018.
17. Clark, R.R. *An Introduction to Dynamical Systems: Continuous and Discrete*; American Mathematical Society: Providence, FL, USA, 2012.
18. Gaston, J. Mémoire sur l'Iteration des Fonctions Rationnelles. *J. Math. Pures Appl.* **1918**, *8*, 47–245.
19. Fatou, P. Sur les Équations Fonctionnelles: Premier Mémoire. *Bull. Soc. Math. Fr.* **1919**, *47*, 161–271. [CrossRef]
20. Fatou, P. Sur les Équations Fonctionnelles: Troisième Mémoire. *Bull. Soc. Math. Fr.* **1920**, *48*, 208–314. [CrossRef]
21. Wang, S.; Xian, H.; Liu, T.; Shateyi, S. Solving Nonlinear Equation Systems via a Steffensen-Type Higher-Order Method with Memory. *Mathematics* **2024**, *12*, 3655. [CrossRef]
22. Chun, C. Construction of Newton-Like Iterative Methods for Solving Nonlinear Equations. *Numer. Math.* **2006**, *104*, 297–315. [CrossRef]

23. Grau-Sánchez, M.; Grau, Á.; Noguera, M. Ostrowski-Type Methods for Solving Systems of Nonlinear Equations. *Appl. Math. Comput.* **2011**, *218*, 2377–2385. [CrossRef]
24. Sharma, J.R.; Guha, R.K.; Sharma, R. An Efficient Fourth Order Weighted-Newton Method for Systems of Nonlinear Equations. *Numer. Algorithms* **2013**, *62*, 307–323. [CrossRef]
25. Arroyo, V.; Cordero, A.; Torregrosa, J.R.; Vassileva, M. Artificial Satellites Preliminary Orbit Determination by Modified High-Order Gauss Methods. *Int. J. Comput. Math.* **2012**, *89*, 347–356. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.